CLAIMS

WHAT IS CLAIMED IS:

1. A method for supporting a platform independent object format for a run-time environment, comprising the computer-implemented steps of:

accessing a definition of an object in terms of a composition of one or more primitive types;

accessing a platform-specific description of layout parameters of the one or more primitive types; and

generating a layout for the object in a high-order language based on the definition of the object and the platform-specific description.

2. The method according to claim 1, further comprising the step of generating instructions for an accessor operation to access a slot in the object holding a value for one of the one or more primitive types.

3. The method according to claim 1, further comprising the step of generating instructions for a get operation to fetch a value for one of the one or more primitive types from a slot in the object.

4. The method according to claim 1, further comprising the step of generating instructions for a set operation to store a value for one of the one or more primitive types from a slot in the object.

5. The method according to claim 1, wherein the one or more primitive types includes or more of the following types: integer, floating point, and reference.

6. The method according to claim 5, wherein the primitive reference type is one of a native machine pointer type and a numeric reference type.

7. The method according to claim 1, wherein the layout parameters include a size and an alignment of the primitive types.

8. A method for supporting an object format for a plurality of incompatible platforms for a run-time environment, comprising the computer-implemented steps of:

accessing a definition of an object as a plurality of slots containing a primitive type;

accessing a plurality of platform-specific descriptions of layout parameters of the one or more primitive types, said platform-specific descriptions corresponding respectively to the incompatible platforms; and

generating a plurality of layouts, corresponding respectively to the incompatible platforms, for the object in a high-order language based on the definition of the object and the platform-specific descriptions.

9. The method according to claim 8, where the slots are located in the layouts for the incompatible platforms, when compiled by a corresponding platform-specific compiler, at same offsets.

10. A computer-readable medium bearing instructions for supporting a platform independent object format for a run-time environment, said instructions being arranged to cause one or more processors upon execution thereby to perform the steps of:

accessing a definition of an object in terms of a composition of one or more primitive types;

accessing a platform-specific description of layout parameters of the one or more primitive types; and

generating a layout for the object in a high-order language based on the definition of the object and the platform-specific description.

11. The computer-readable medium according to claim 10, wherein said instructions are further arranged for performing the step of generating instructions for an accessor operation to access a slot in the object holding a value for one of the one or more primitive types.

12. The computer-readable medium according to claim 10, wherein said instructions are further arranged for performing the step of generating instructions for a get operation to fetch a value for one of the one or more primitive types from a slot in the object.

13. The computer-readable medium according to claim 10, wherein said instructions are further arranged for performing the step of generating instructions for a set operation to store a value for one of the one or more primitive types from a slot in the object.

14. The computer-readable medium according to claim 10, wherein the one or more primitive types includes or more of the following types: integer, floating point, and reference.

15. The computer-readable medium according to claim 14, wherein the primitive reference type is one of a native machine pointer type and a numeric reference type.

16. The computer-readable medium according to claim 10, wherein the layout parameters include a size and an alignment of the primitive types.

17. A computer-readable medium bearing instructions for supporting an object format for a plurality of incompatible platforms for a run-time environment, said instructions being arranged to cause one or more processors upon execution thereby to perform the steps of:

accessing a definition of an object as a plurality of slots containing a primitive type;

accessing a plurality of platform-specific descriptions of layout parameters of the one or more primitive types, said platform-specific descriptions corresponding respectively to the incompatible platforms; and

generating a plurality of layouts, corresponding respectively to the incompatible platforms, for the object in a high-order language based on the definition of the object and the platform-specific descriptions.

18. The computer-readable medium according to claim 17, wherein the slots are located in the layouts for the incompatible platforms, when compiled by a corresponding platform-specific compiler, at same offsets.